IN THE CLAIMS

Please amend claims 5-7 and add new claims 14 and 15, as shown:

1. (Original) A method of non-centralized zero-knowledge authentication for a computer network, comprising steps of:

establishing a first computer having a first authentication agent and a first prover agent on the computer network;

detecting a first authentication request over the computer network from a second computer having a second prover agent;

authenticating the second prover agent through a zero-knowledge identification protocol; and

promoting the second computer with a second authentication agent to perform authentication for the computer network.

2. (Previously Presented) The method of claim 1, further comprising periodically generating and distributing a new secret to the first and second authentication agents.

3. (Original) The method of claim 1, further comprising:

detecting a second authentication request over the computer network from a third computer having a third prover agent;

authenticating the third prover agent through a zero-knowledge identification protocol with the second authentication agent; and

promoting the third computer with a third authentication agent to perform authentication for the computer network.

4. (Previously Presented) The method of claim 1, further comprising periodically publishing encrypted numbers for the zero-knowledge identification protocol, including the steps of:

generating first and second large prime numbers;

calculating a product of the first and second large prime numbers;

generating a secret to have a value relatively prime to the product, greater than zero and less than the product;

encrypting the product;

encrypting the secret; and

publishing encrypted values of the secret and product.

5. (Currently Amended) A method of protecting a host computer from unauthorized ~~client~~ access by a client computer over a computer network, comprising the steps of:

installing a prover agent application on the client computer;

installing a verifier agent application on the host computer;

creating a trusted source application on the computer network to generate and publish encrypted values of a secret and product of first and second large prime numbers;

reading the encrypted values for the secret and product, by the prover and verifier from the trusted source;

decrypting the secret, by the prover and verifier;

decrypting the product, by the prover and verifier; and

performing a plurality of verification dialog between the prover and verifier over the network, wherein the prover demonstrates knowledge of the secret and product without exposing the values of the secret and product, and wherein the client is denied access to a secure area of the host when the prover fails to demonstrate knowledge of the secret and product and granted access to the secure area when the client succeeds in demonstrating knowledge of the secret and product.

6. (Currently Amended) The method of claim 5, wherein the steps of decrypting the secret and product further utilize previous values of the secret and product as operators in the modulus inverse operations, to decrypt new values for the secret and the product.

7. (Currently Amended) The method of claim 5, further comprising:

installing a first agent to be authenticated on a third computer on the network, the first agent having values for s, n and t, s being the secret, n being the product, and t being a size of an answer set;

installing a second agent on a fourth computer on the network, to authenticate the first agent, the second agent having values for s, n, and t;

generating r as a random number generated by the first agent;

calculating x by the first agent, r being raised to power of t modulus n;

sending x from the first agent to the second agent, over the network;

calculating b by the second agent, b being further defined as a member of set of integers from zero through t-1;

sending b from the second agent to the first agent, over the network;

calculating y by the first agent, y being a product of r*s raised to power of b;

sending y from the first agent to the second agent, over the network; and

determining authentication of the first agent, by determining equivalence of a first equation to a second equation, if y is not equal to zero, first equation is y^t mod n and second equation is (xv^b) mod n.

8. (Original) A system of non-centralized zero-knowledge authentication for a computer network, comprising:

two or more computers establishing the computer network, each of the computers containing an authentication agent, secret and prover agent; and

a requesting computer having a prover agent, for requesting access to the computer network,

wherein the prover agent of the requesting computer and one of the authentication agents of the two or more computers engaging in a zero-knowledge authentication protocol, and wherein the requesting computer operates with an authentication agent on the computer network when the requesting computer is authenticated through the zero-knowledge authentication protocol.

9. (Original) The system of claim 8, further comprising a trusted source for periodically generating a new secret for the authentication agents of computers on the network.

10. (Original) The system of claim 8, the requesting computer comprising a cell phone.

11. (Previously Presented) The system of claim 8, the computer network comprising one or more of the Internet, a local area network, a communications link, and a wireless network.

12. (Original) The system of claim 8, the authentication agents and prover agents being installed on each of the computers through common software.

13. (Original) A software product comprising instructions, stored on computer-readable media, wherein the instructions, when executed by a computer, perform steps for non-centralized zero-knowledge authentication for a computer network, comprising:

instructions for establishing a first computer having a first authentication agent and a first prover agent on the computer network;

instructions for detecting a first authentication request over the computer network from a second computer having a second prover agent;

instructions for authenticating the second prover agent through a zero-knowledge identification protocol; and

instructions for promoting the second computer with a second authentication agent to perform authentication for the computer network.

14. (New) The method of claim 5, wherein the prover has values for s, n and t, s being the secret, n being the product, and t being a size of an answer set and wherein the verifier having values for s, n and t; the verification dialog between the prover and verifier including:

generating r as a random number by the prover agent;

calculating x by the prover agent, r being raised to power of t modulus n;

sending x from the prover agent to the verifier agent, over the network;

calculating b by the verifier agent, b being further defined as a member of set of integers from zero through t-1;

sending b from the verifier agent to the prover agent, over the network;

calculating y by the prover agent, y being a product of r*s raised to power of b;

sending y from the prover agent to the verifier agent, over the network; and

determining authentication of the prover agent, by determining equivalence of a first equation to a second equation, if y is not equal to zero, the first equation is y^t mod n and the second equation is (xv^b) mod n.

15. (New) A method of protecting a host computer from unauthorized access over a computer network, comprising the steps of:

installing a prover agent application on a client computer;

installing a verifier agent application on the host computer;

creating a trusted source application on the computer network to generate and publish encrypted values of a secret and product of first and second large prime numbers;

reading the encrypted values for the secret and product, by the prover and verifier from the trusted source;

decrypting the secret, by the prover and verifier;

decrypting the product, by the prover and verifier;

performing a plurality of verification dialog between the prover and verifier over the network, wherein the prover demonstrates knowledge of the secret and product without exposing the values of the secret and product, and wherein the client is denied access to a secure area of the host when the prover fails to demonstrate knowledge of the secret and product and granted access to the secure area when the client succeeds in demonstrating knowledge of the secret and product;

installing a first agent to be authenticated on a third computer on the network, the first agent having values for s, n and t, s being the secret, n being the product, and t being a size of an answer set;

installing a second agent on a fourth computer on the network, to authenticate the first agent, the second agent having values for s, n, and t;

generating r as a random number generated by the first agent;

calculating x by the first agent, r being raised to power of t modulus n;

sending x from the first agent to the second agent, over the network;

calculating b by the second agent, b being further defined as a member of set of integers from zero through t-1;

sending b from the second agent to the first agent, over the network;

calculating y by the first agent, y being a product of r*s raised to power of b;

sending y from the first agent to the second agent, over the network; and

determining authentication of the first agent, by determining equivalence of a first equation to a second equation, if y is not equal to zero, first equation is y^t mod n and second equation is (xv^b) mod n.